**ACNet-based MobileNet for image classification**

# Abstract

In this paper, we propose a novel ACNet-based MobileNet(Adaptively Connected Neural Networks based MobileNet) for image classification. Google's MobileNet[1] gets a significant achievement in image classification on the mobile device platform in recent years. However, MobileNet has fewer model parameters, making its accuracy still not comparable to other large-scale network models. Previously, ACNet[2] proposed to improve the traditional convolutional neural networks (CNNs), can flexibly change the global and local reasoning in the internal feature performance, and it also enhances classification accuracy. We believe that ACNet can adequately compensate for the above-mentioned MobileNet problems. Therefore, our ACNet – based MobileNet has benefited is that while retaining the inverted residual architecture of the MobileNet model, the model parameters are small enough. It also could improve the accuracy of image classification slightly. The code is available at *https://github.com/TOMMYWHY/acnet\_mobilenetv3*

# Introduction

# Background and Literature review

# Methodology

**Architecture**

Guangrun Wang proposed the Acnet mention formula 1 which uses the weights of β and γ to control CNN and MLP adaptively. It makes a specific layer of the model have both global inference and local inference. The formula also mentions a weight α, which is the weight that controls its transformation.

Inspired by ACnet, I proposed formula 2 which contains two parts of global reasoning and self-transformation and is controlled by weights γ and α, respectively. When the formula 2 applied to the 1\*1 convolution operation, it can change the dimension and at the same time has the ability of global inference.

By analyzing the network structure of MobileNet, the 1\*1 convolution in the unique inverted residual design will lose too much information. The ACNet-based MobileNet I proposed uses formula 2 to optimize the 1\*1 convolution operation in the inverted residual module. It makes the inverted residual module have certain global inference ability. X is the input layer, Uij​,Wij​, represent the learnable weights. Use the local transformation or the global transformation for self-inference through the three weight parameters of α,γ. The convolution operation can adaptively learn the weight parameters through backpropagation.

We can obtain an adaptive neural layer through the above formula, and then apply the layer model to the inverted residual module (Figure1) to obtain an adaptive inverted residual module(AIR). Universal adaptive inverted residual module to adaptively capture global and local dependencies. So that ACNet-based MobileNet has higher accuracy and global reasoning ability.

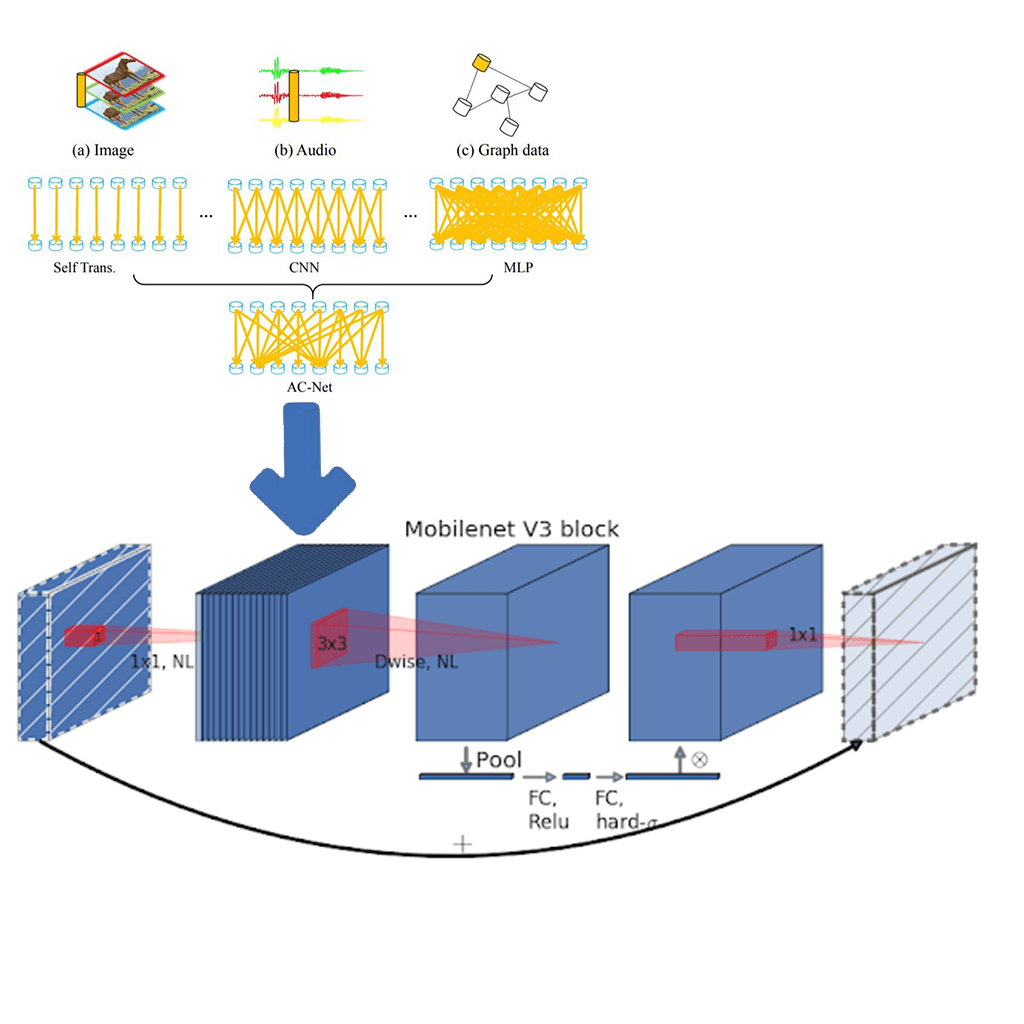


Figure1: the inverted residual with linear bottleneck

**Implementation**

Mobilenetv3 uses the inverted residual with linear bottleneck module and the structure of squeeze and excitation. We propose an adaptive inverted residual module (AIR) with the same structure. In Figure 2 AIR is defined by a 1x1 Expansion layer followed by depth-wise convolutions and a 1x1 projection layer. MnasNet[11] built upon the MobileNetV2 structure by introducing lightweight attention modules based on squeeze and excitation into the bottleneck structure. The Expansion layer and Projection layer use its change deformation and global optimization.



Figure2: adaptive inverted residual module (AIR)

# Experiments

**Dataset**

This article uses Cifar-100[12] dataset. The dataset has 100 classes, and each class includes 600 images. Among the 60000 images, there are 50000 training images and 10000 testing images. The reason for using cifar100 as the data set this time is to verify that ACNet-based MobileNet has a certain global inference ability through more detailed classification. According to a convention, two error rates will be provided: top-1 and top-5. The top-5 error rate means that the testing image's correct label is not among the five most likely notes considered by the model. In the data preprocessing stage, the data pictures are uniformly cropped to a fixed size of 224\*224 at the center point. We will mirror and flip the image to achieve the purpose of expanding the dataset. All the implementation details and experiment settings are the same as [13][14].

**Training**

We use multiple controlled trials to test ACNet-based MobileNet. The first is to verify the validity of the ACNet-based MobileNet theory. Use the MobileNetV3-Small structure mentioned in the mobilenetv3 paper for training on cifar100. Then use ACNet-based MobileNet (Table1)with the same structure and params to classify and predict cifar100. Use a single factor to verify the functionality of ACNet-based MobileNet. At the same time, we used the techniques in Tong's[15] paper to optimize the model parameters and tried to get the model with the highest accuracy.

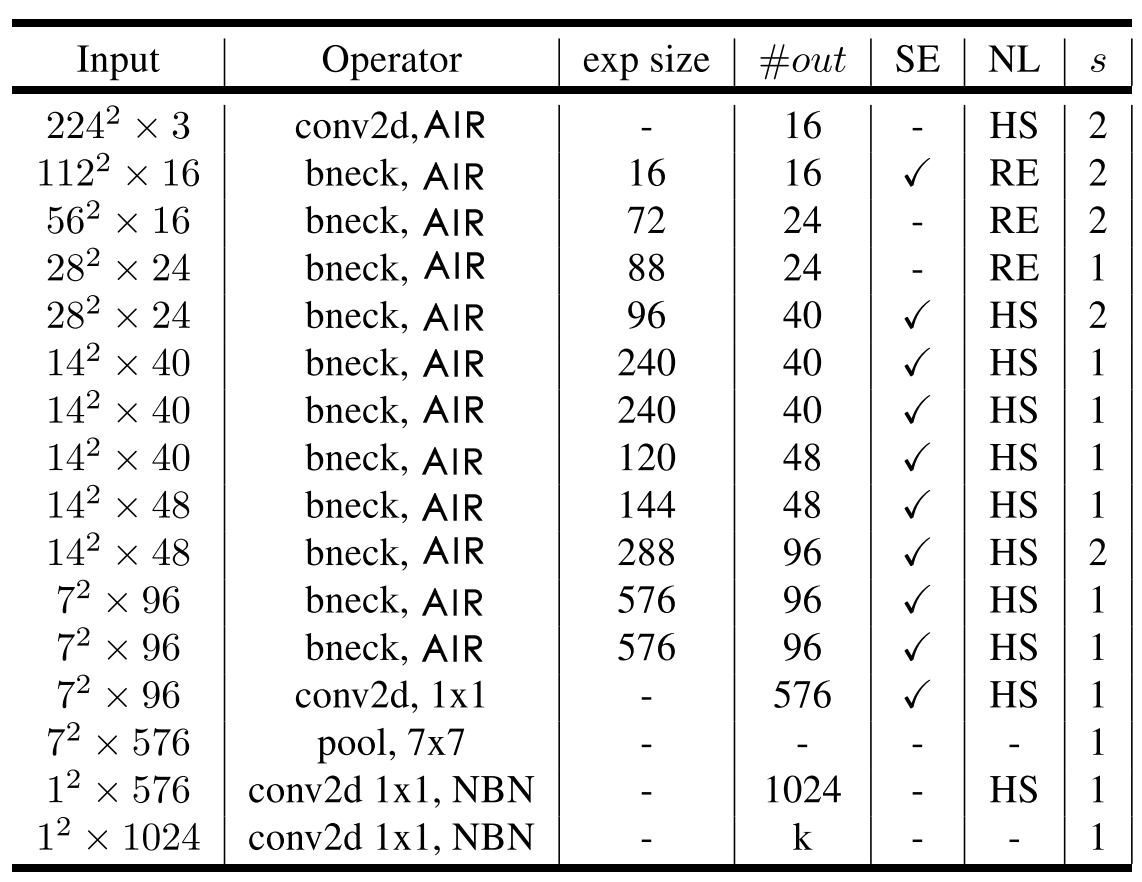


Table1: It uses the same architecture as mobilenetv3, which uses AIR instead of 3X3 and 5X5 convolution operations.

**﻿**

**Experiment results**

I compared the experimental results of epoch 55 and 100. It can be seen from the experimental results that the accuracy of ACNet-based MobileNet is a little bit higher than that of MobileNet V3. And the training time of the model have been improved.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  | **acc1** | **acc5** | **Epoch** | **Training Time** |
| MobileNetV3(SMALL) | 66.02 | 89.12 | 55 | 1h21min |
| **ACNet-based MobileNet** | **66.59** | **89** | 55 | 1h55min |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  | **acc1** | **acc5** | **Epoch** | **Training Time** |
| MobileNetV3(SMALL) | 66.6 | 89.25 | 100 | 2h10min |
| **ACNet-based MobileNet** | **66.95** | **89.27** | 100 | 2h53min |

Table 2: ACNet-based MobileNet results

It can be found from the above experimental results that the accuracy of ACNet-based MobileNet is not as expected. And the training time of the model has also increased as well. From the two sets of experiments, it can be seen that there is not much difference between the results of 55 iterations and 100 iterations. It may be caused by setting the hyperparameter learning rate too low and stopping at the saddle point. The learning rate needs to be searched to get a better model.

# conclusions

The ACNet-based MobileNet model uses the AIR module, which in theory can effectively improve the global reasoning ability and retain more information in the inverted residual module. However, it has not been significantly confirmed in this experiment. Try to use the ACNet-based MobileNet model to distinguish objects with the same shape characteristics effectively, and then get a more accurate model. From the experimental results, the effect is not significant. The reason may be the hyperparameter learning rate is set too small, and follow-up work will conduct search experiments on the learning rate to obtain a better performing model. In the next work, other different data sets will be used to verify the model to ensure the rigor of the conclusion.

The results of the experiment did not meet expectations. However, we still got a conclusion that the method of adaptively adjusting MLP and CNN cannot solve the information loss caused by the dimensional changes in the inverted residual module. The author speculates that the reason is the inverted residual module is reduced to the loss of two-dimensional information, which cannot be compensated by simple global inference. In the follow-up work, a lot of experiments and explorations will be carried out to find an innovative inverted residual module that retains most of the information as much as possible. In this way, image classification technology can be better applied to the mobile terminal, and deep learning related technology can be better applied to life.

# References

[1] A. Howard *et al.*, “Searching for mobileNetV3,” *Proc. IEEE Int. Conf. Comput. Vis.*, vol. 2019-Octob, pp. 1314–1324, 2019, doi: 10.1109/ICCV.2019.00140.

[2] G. Wang, K. Wang, and L. Lin, “Adaptively connected neural networks,” *Proc. IEEE Comput. Soc. Conf. Comput. Vis. Pattern Recognit.*, vol. 2019-June, pp. 1781–1790, 2019, doi: 10.1109/CVPR.2019.00188.

[3] A. G. Howard and W. Wang, “MobileNets: Efficient Convolutional Neural Networks for Mobile Vision Applications,” 2012.

[4] M. Sandler, A. Howard, M. Zhu, A. Zhmoginov, and L. C. Chen, “MobileNetV2: Inverted Residuals and Linear Bottlenecks,” *Proc. IEEE Comput. Soc. Conf. Comput. Vis. Pattern Recognit.*, pp. 4510–4520, 2018, doi: 10.1109/CVPR.2018.00474.

[5] K. He, X. Zhang, S. Ren, and J. Sun, “Deep residual learning for image recognition,” *Proc. IEEE Comput. Soc. Conf. Comput. Vis. Pattern Recognit.*, vol. 2016-Decem, pp. 770–778, 2016, doi: 10.1109/CVPR.2016.90.

[6] K. Simonyan and A. Zisserman, “Very deep convolutional networks for large-scale image recognition,” *3rd Int. Conf. Learn. Represent. ICLR 2015 - Conf. Track Proc.*, pp. 1–14, 2015.

[7] Y. LeCun, L. Bottou, Y. Bengio, and P. Haffner, “Gradient-based learning applied to document recognition,” *Proc. IEEE*, vol. 86, no. 11, 1998, doi: 10.1109/5.726791.

[8] A. Krizhevsky, I. Sutskever, and G. E. Hinton, “ImageNet classification with deep convolutional neural networks,” *Commun. ACM*, vol. 60, no. 6, pp. 84–90, 2017, doi: 10.1145/3065386.

[9] C. Szegedy *et al.*, “Going deeper with convolutions,” in *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, 2015, vol. 07-12-June-2015, doi: 10.1109/CVPR.2015.7298594.

[10] O. Russakovsky *et al.*, “ImageNet Large Scale Visual Recognition Challenge,” *Int. J. Comput. Vis.*, vol. 115, no. 3, pp. 211–252, 2015, doi: 10.1007/s11263-015-0816-y.

[11] M. Tan *et al.*, “Mnasnet: Platform-aware neural architecture search for mobile,” *Proc. IEEE Comput. Soc. Conf. Comput. Vis. Pattern Recognit.*, vol. 2019-June, pp. 2815–2823, 2019, doi: 10.1109/CVPR.2019.00293.

[12] M. B. McCrary, “Urban multicultural trauma patients.,” *ASHA*, vol. 34, no. 4, 1992.

[13] G. Wang, P. Luo, and X. Wang, “Batch Kalman Normalization: Towards Training Deep Neural Networks with Micro-Batches.”

[14] G. Wang, J. Peng, P. Luo, X. Wang, and L. Lin, “Kalman normalization: Normalizing internal representations across network layers,” *Adv. Neural Inf. Process. Syst.*, vol. 2018-December, no. NeurIPS, pp. 21–31, 2018.

[15] T. He, Z. Zhang, H. Zhang, Z. Zhang, J. Xie, and M. Li, “Bag of tricks for image classification with convolutional neural networks,” *Proc. IEEE Comput. Soc. Conf. Comput. Vis. Pattern Recognit.*, vol. 2019-June, pp. 558–567, 2019, doi: 10.1109/CVPR.2019.00065.